BANCO CENTRAL DE COSTA RICA
DIVISIÓN ECONÓMICA
DEPARTAMENTO DE CONTABILIDAD SOCIAL
DCS-155-2005-
SETIEMBRE DEL 2005

PRONÓSTICO DE INFLACIÓN EN COSTA RICA: UNA ESTIMACIÓN CON REDES NEURONALES ARTIFICIALES

Álvaro Solera Ramírez

Documento de trabajo del Banco Central de Costa Rica, elaborado en el
Departamento de Contabilidad Social

Las ideas expresadas en este documento son responsabilidad del autor y no necesariamente
representan la opinión del Banco Central de Costa Rica

TABLA DE CONTENIDO

1. INTRODUCCIÓN....4

2. ARQUITECTURA DE REDES NEURONALES ARTIFICIALES....5

2.1 VISIÓN GENERAL....5

2.2 MODELO DE UNA NEURONA BIOLÓGICA....6

2.3 MODELO MATEMÁTICO SIMPLE DE UNA NEURONA ARTIFICIAL....7

2.4 FASE DE APRENDIZAJE Y DE TRABAJO DE UNA RED NEURONAL ARTIFICIAL....8

2.5 APLICACIONES DE REDES NEURONALES ARTIFICIALES EN EL ANÁLISIS DE SERIES DE TIEMPO....9

2.6 INTERPRETACIÓN ECONÓMICA DE LAS SUPERFICIES ESCONDIDAS....10

3. RESULTADOS....11

4. CONCLUSIONES....14

5. REFERENCIAS....16

6. ANEXOS....18

6.1 FIGURAS....18

6.2 PROGRAMACIÓN EN RATS PARA LA ESTIMACIÓN DE REDES NEURONALES ARTIFICIALES....21

PRONÓSTICO DE INFLACIÓN EN COSTA RICA: UNA ESTIMACIÓN CON REDES NEURONALES ARITIFICIALES

Resumen

En este documento se construyen tres modelos de pronóstico para la inflación aplicando la técnica de Redes Neuronales Artificiales (RNA). Esta técnica de modelación es capaz de capturar relaciones no lineales entre las variables, con lo cual se constituye en una poderosa alternativa en técnicas de regresión estándar para modelar y pronosticar series de tiempo. En la modelación de variables económicas por lo general se asume una relación lineal. El supuesto de linealidad, sin embargo, no se justifica cuando se identifican efectos asimétricos sobre el nivel de precios de la política monetaria.

De los modelos elaborados dos utilizan como insumo agregados monetarios esperando que éstos contengan información útil acerca de la inflación futura, dado que existe consenso en que la inflación en el largo plazo tiene un origen monetario.

Los modelos de RNA tienen un mejor desempeño, en cuanto a la precisión del pronóstico, que los modelos de periodicidad mensual que ha venido actualizando la División Económica para pronosticar la inflación. Únicamente el modelo denominado Ingenuo no es superado. Por otra parte, se amplia la gama de posibles modelos para analizar la evolución futura de la inflación y se incluyen variables monetarias, ausentes en los otros modelos de pronóstico.

FORECASTING INFLATION IN COSTA RICA USING ARTIFICIAL NEURAL NETWORKS

Abstract

This paper presents three models to forecast inflation using Artificial Neural Networks (ANN). The ANN method captures non-linearities between variables, therefore it could be a powerful alternative to standard regression and time series forecasting. When modeling economic variables is often assumed that there are linear relationships. However, the argument of linearity cannot be justified when it is known that monetary policy has asymmetric effects on the price level.

Since there is consensus that long run inflation has a monetary origin, two of the models elaborated include monetary aggregates hoping that they will contain useful information about future inflation.

The ANN models have shown better performance in forecasting inflation than those monthly updated models that the Economic Division have been using. The model called Ingenue is the only one which is not excelled. On the other hand, the number of models to analyze future inflation is increased and monetary variables which were absent in other forecasting models are incorporated.

Clasificación JEL: E31, E37. E52

1. INTRODUCCIÓN

Este documento se enmarca dentro de la investigación desarrollada por el Banco Central de Costa Rica para pronosticar la inflación. Mejorar y complementar los ejercicios de pronóstico de la inflación son de suma importancia para el diseño e implementación de la política monetaria y, en consecuencia para el logro del objetivo esencial de estabilidad interna y externa de la moneda nacional encomendado a la Institución.

Existe consenso en que la inflación, en el mediano y largo plazo, tiene un origen estrictamente monetario, razón por la cual es de esperar que los agregados monetarios contengan información útil acerca de la dirección futura de la inflación. Por otra parte, los trabajos de Friedman (1968) y Cover (1992) confirman la asimetría[1] entre la política monetaria y el nivel de precios[2]. Así, el supuesto generalmente aceptado de una relación lineal en la modelación de variables económicas podría no justificarse entre algunas, dada la evidencia empírica.

Dadas las premisas anteriores, el objetivo de este documento es establecer la relación entre dinero e inflación por medio de un modelo de redes neuronales artificiales (RNA). Esta técnica de modelación, capaz de capturar no linealidades, tiene ventajas en comparación con otros métodos tradicionales de pronóstico. Entre las ventajas que destaca Serju (2001) se encuentran: i) aproximar la forma funcional que mejor caracteriza las series de tiempo y ii) superar los métodos de pronóstico tradicionales en horizontes de largo plazo, y ser tan buena como éstos en el pronóstico de corto plazo[3].

La hipótesis que subyace en la modelación por medio de RNA es que las relaciones entre variables pueden ser mejor capturadas por medio de modelos no lineales. Desde este punto de vista se constituye en una poderosa alternativa en técnicas de regresión estándar para modelar y pronosticar series de tiempo[4]. El amplio desarrollo de este enfoque en series de tiempo se atribuye a la capacidad que posee para permitir relaciones univariables o multivariables no lineales muy generales, que podrían ser difíciles de ajustar usando otras técnicas[5] y que representan elementos esenciales para el pronóstico y modelación de las variables.

---

[1] La asimetría se origina cuando la economía responde menos a estímulos de política positivos que ante estímulos negativos o viceversa.

[2] En Morgan (1993) y Martínez (2003) también encuentran evidencia a favor de efectos asimétricos de la política monetaria sobre la producción real.

[3] En Tkacz (2000) los resultados sugieren que para horizontes de mediano y largo plazo los modelos no lineales dominan a los lineales, en la determinación de la relación entre diferencias de rendimientos de las tasas de interés y cambios en la tasa de inflación.

[4] Se encuentran desarrollos aplicados a los precios de las acciones, el tipo de cambio, las tasas de interés, el producto y la inflación.

[5] Este enfoque también permite la modelación de variables en donde la presencia de un modelo estructural no es clara, dado que no parte de supuestos a-priori sobre los datos para el pronóstico y todo lo que de ellas puede derivarse es inherente a las observaciones (Evans, 1997).

Como se destaca en Misas et al. (2002) "en el contexto de análisis de series de tiempo este enfoque se clasifica dentro de los modelos no lineales y su atractivo se encuentra en: i) su habilidad para aprender los rasgos o características subyacentes en los datos, es decir, para identificar dependencias con base en una muestra finita, de manera que el conocimiento adquirido pueda ser generalizado a muestras no observadas, ii) que se ha constituido en un marco de modelación poderoso e interesante cuyo potencial ha sido comprobado en aplicaciones en diversas ciencias, iii) que este enfoque no está sujeto a supuestos restrictivos como la linealidad, que suele ser necesaria para la aplicación de los modelos matemáticos tradicionales".

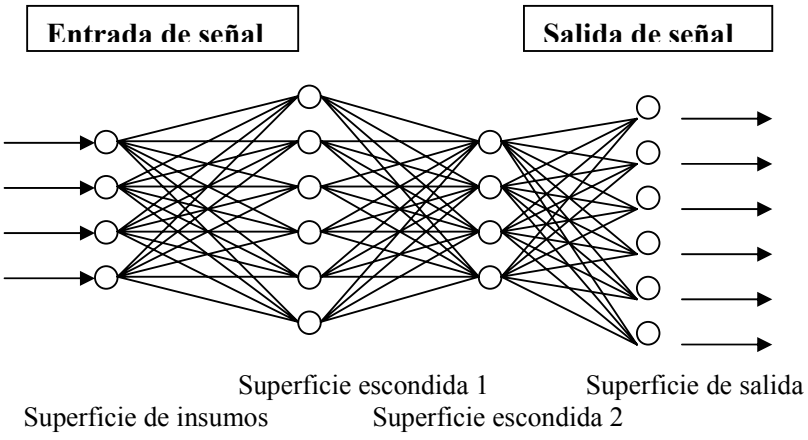
El resto del documento consiste de dos secciones. La segunda, describe la arquitectura de las redes neuronales artificiales. La tercera sección presenta los resultados de la aplicación práctica efectuada y finalmente en la última se presentan las conclusiones.

2. ARQUITECTURA DE REDES NEURONALES ARTIFICIALES

2.1 VISIÓN GENERAL

La imitación de la mente humana por medio de un software electrónico ha sido la meta de científicos por décadas. Hace alrededor de 50 años que los investigadores crearon el primer software para modelar el funcionamiento de las células nerviosas. Desde entonces, un gran segmento de la comunidad científica continúa trabajando sobre nuevos algoritmos y modelos matemáticos. Hoy en día, las denominadas redes neuronales artificiales (RNA) dominan el interés en este campo. Las RNA utilizan una cantidad de simples unidades computacionales llamadas "neuronas", cada una de las cuales trata de imitar las funciones básicas de la neurona biológica.

Figura 1



La figura 1 muestra la estructura básica de una red neuronal artificial. Cada neurona en la red recibe y envía señales. Esta comunicación entre neuronas se realiza por medio de conectores, los cuales tienen asociada una determinada ponderación[6].

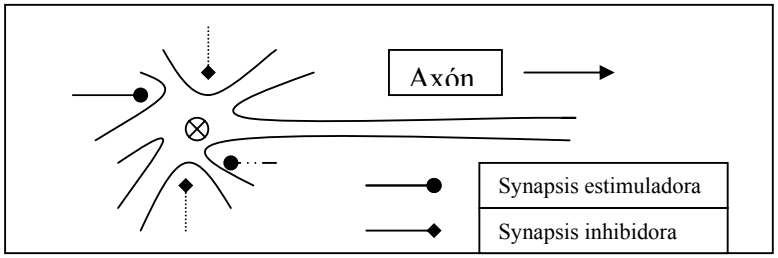
Las neuronas que ingresan señales (inputs) a la red constituyen la superficie de insumos, mientras que las neuronas que tienen el resultado (output) de la red conforman la superficie de salida. Estas dos superficies son el contacto con el mundo exterior. Adicionalmente, entre la superficie de insumos y de salida pueden existir una o más superficies, generalmente denominadas superficies escondidas (1 y 2 en la figura 1), que también procesan la información pero no tienen contacto con el mundo exterior. Así, cuando la superficie de insumos recibe una señal sus neuronas producen un resultado, el cual se convierte en señal para la siguiente superficie. Este proceso continúa hasta que ciertas condiciones son alcanzadas y la superficie de salida produce el resultado final de la red.

El objetivo de la red neuronal es desarrollar un algoritmo[7] mecánico de aprendizaje que lleve a aprender la solución del problema, a través de la modificación de la ponderación de los conectores dentro de la red. Entre los algoritmos de aprendizaje el más utilizado es el de propagación hacia atrás del error, en el cual la información sobre los errores (la diferencia entre los resultados de la red y los valores observados) es filtrada y retornada al sistema para ajustar las ponderaciones de los conectores entre las diferentes superficies y de esta forma incrementar el desempeño de la red.

2.2 MODELO DE UNA NEURONA BIOLÓGICA

En el área de redes neuronales biológicas los investigadores han analizado varios modelos para explicar cómo funcionan las células del cerebro humano[8]. Como se observa en la figura 2 las células contienen un núcleo y una membrana externa eléctrica. Cada neurona tiene un nivel de activación, con rangos entre un máximo y un mínimo.

Figura 2



[6] Estas cumplen el mismo propósito que los coeficientes de un modelo de regresión. Dado un conjunto de valores para los insumos (variables explicativas) los ponderadores determinan el resultado de la red.
7 Se entiende por algoritmo un conjunto específico de instrucciones para resolver un procedimiento o problema, usualmente con el requerimiento de que el procedimiento termina en algún punto.
[8] El cerebro humano contiene alrededor de 1011 células nerviosas con aproximadamente 1014 conexiones con otras células.

El aumento o disminución de la activación de una neurona producto de la relación con otra se da por medio de la sinapsis[9] o espacio de comunicación. Esta sinapsis traslada el nivel de activación desde una neurona emisora a una receptora. Si la sinapsis es del tipo estimuladora, el nivel de activación proveniente de la neurona emisora incrementa la activación de la neurona receptora, caso contrario cuando la sinapsis es del tipo inhibidora. La sinapsis no solo difiere por el hecho de ser estimuladora o inhibidora sino también por la intensidad del estímulo. El resultado de cada neurona es transferido por el axón para influir sobre otras neuronas.

Este simple modelo neuronal biológico subyace en la mayoría de las aplicaciones de RNA en la actualidad. Aunque el modelo es únicamente una aproximación muy gruesa de la realidad, sus exitosas aplicaciones han probado los beneficios de las redes neuronales.

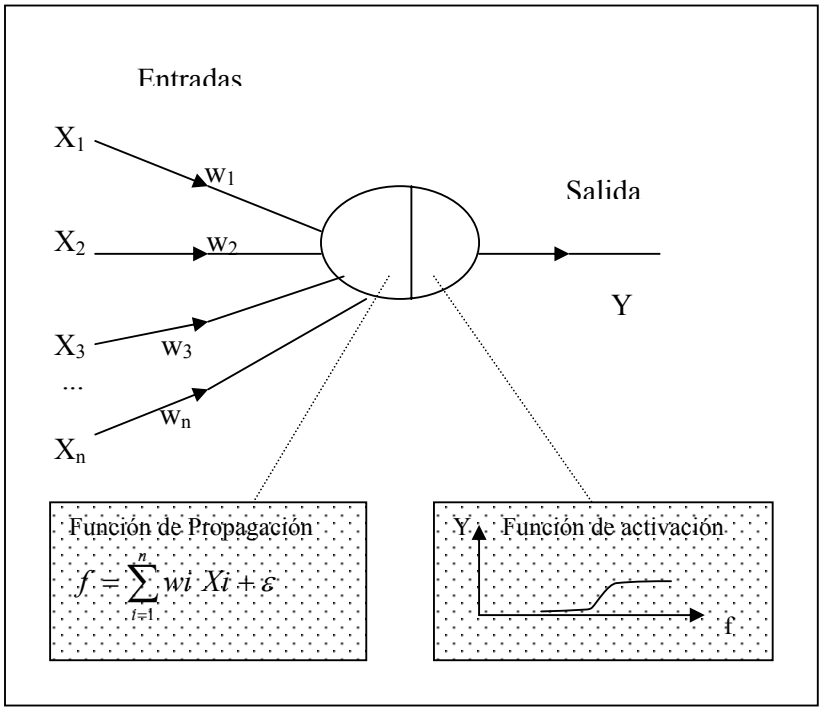
2.3 MODELO MATEMÁTICO SIMPLE DE UNA NEURONA ARTIFICIAL

Varios modelos matemáticos se han construido sobre la base del modelo neuronal biológico. Uno de los más comunes se denomina función de propagación el cual combina todas las entradas (insumos) X_i que provienen desde las neuronas emisoras. El método de combinación es una suma ponderada, donde los ponderadores W_i representan la potencia (intensidad) de la sinapsis. Las sinapsis estimuladoras tienen ponderaciones positivas mientras que las inhibidoras tienen ponderadores negativos. Para expresar completamente el nivel de activación de una neurona, un error es sumado a la suma ponderada.

La figura 3 muestra un simple modelo matemático de una neurona artificial. Todas las entradas son combinadas por una suma ponderada (función de propagación). Entonces el nivel de activación de una neurona es calculado por la función de activación. El nivel de activación es también la señal de salida. La función de activación calcula la señal de salida Y de la neurona proveniente del nivel de activación f.

[9] Zona en la que se unen dos neuronas.

Figura 3



2.4 FASE DE APRENDIZAJE Y DE TRABAJO DE UNA RED NEURONAL ARTIFICIAL

El primer paso en el diseño de una solución para la RNA es la fase de aprendizaje. Para ello se utiliza ya sea un conjunto de datos provenientes de una muestra o un "maestro". Este último es una función matemática o una persona que evalúa la calidad del desempeño de la red. Dado que las redes neuronales son usadas generalmente en aplicaciones complejas donde no existen buenos modelos matemáticos o la evaluación del desempeño es difícil, la mayoría de las aplicaciones usan los datos de la muestra en el proceso de aprendizaje.

Finalizada la fase de aprendizaje, la red está lista para ser usada. Esto se denomina la fase de trabajo. Como resultado del aprendizaje o entrenamiento, la red neuronal producirá resultados con base en los datos incluidos en la muestra. En esta fase, el comportamiento de la red es determinístico. Esto es, para cada combinación de valores de insumos, el valor de salida siempre será el mismo. Durante la fase de trabajo la red no aprende.

2.5 APLICACIONES DE REDES NEURONALES ARTIFICIALES EN EL ANÁLISIS DE SERIES DE TIEMPO

En Misas et al. (2002) se define la RNA, dentro del contexto de series de tiempo, como modelos no lineales[10] entrenados para (i) realizar conexiones entre los valores pasados y presentes de una serie de tiempo y (ii) extraer estructuras y relaciones escondidas que gobiernan el sistema de información.

La capacidad de las RNA en identificar patrones de comportamiento, en especial no lineales, le permite en el estudio de series de tiempo, poder detectar las dinámicas no lineales y lograr un mejor ajuste en comparación con modelos lineales (Franses y Van Dijk,2000).

Por otra parte, como lo demuestra Tckaz et al. (1999) las RNA tienen la capacidad de reconocer y modelar comportamientos atípicos, entre ellos outliers o cambios en el nivel de las series de tiempo, lo cual indica un alejamiento del mundo lineal.

En particular, la RNA se puede expresar de la siguiente manera:

$$y_t = F(X_t;\Theta) \quad \text{.....(1)}$$

donde:

$$F(X_t;\Theta) = \sum_{k=1}^{Q} \beta_k g(\gamma_k Z_t) + \varepsilon_t \quad \text{.....(2)}$$

con $Z_t \subseteq X_t$, $X_t = \{y_{t-1},...,y_{t-p},w_1,...,w_m\}$ con $y_{t-j}, j = 1,...p$ rezagos de la variable dependiente, w_j, $j = 1,...,m$ variables exógenas y g(u) la función logística, es decir:

$$g(u) = \frac{1}{1+e^{-u}} \quad \text{.....(3)}$$

sirve como una función de "achatamiento" para el valor μ, con lo cual la escala del resultado se encuentra entre cero y uno. Esta función es la función de activación de la red neuronal. Por otra parte, γ_k es el conector (ponderador) entre Z_t y la unidades escondidas k, Q es el número total de unidades escondidas o componentes logísticos[11] y β_k es la ponderación de una unidad k sobre el resultado y_t.

[10] La modelación no lineal junto con la lineal se enmarcan dentro de la modelación paramétrica la cual supone una forma funcional específica para $f()$, usualmente con parámetros que deben ser estimados, para relacionar a una variable con su propia historia y/o con la historia de otras variables.

[11] Se debe tomar en cuenta que muchas unidades escondidas llevan a un sobre aprendizaje, esto origina que la red no aprenda una solución general; por otra parte, pocas unidades escondidas no permiten aprender el patrón de comportamiento entre el insumo y el resultado.

La estimación de los parámetros o conectores (γ_k, β_k) de la ecuación 2 se realiza minimizando la suma de desviaciones al cuadrado entre el resultado observado y el resultado producido por la red neuronal de tal forma que se minimice la siguiente ecuación:

$$SDC = \sum_{t=1}^{T}\left[y_t - \sum_{k=1}^{Q} \beta_k g(\gamma_k Z_t) + \varepsilon_t \right]^2 \quad .....(4)$$

La ecuación 4 es estimada usando el algoritmo de propagación hacia atrás del error, el cual actualiza los valores de los parámetros hasta que se alcanza un nivel de convergencia preestablecido, generalmente este nivel de convergencia se asocia con lograr un valor predeterminado para el coeficiente de determinación (R^2).

En la modelación de series de tiempo por medio de RNA el proceso de aprendizaje o entrenamiento se define como un proceso de iteración y ajuste de parámetros que permite llevar el resultado producido por la RNA, en línea con el resultado observado de forma que se minimice la ecuación 4 (Moshiri y Cameron, 1998).

Por último, en Misas et al. (2002) se destacan limitaciones en la modelación de relaciones entre variables con base en RNA: i) dificultad en la interpretación de los parámetros, ii) no queda claro el modelo paramétrico no lineal que describe los patrones o dinámicas no lineales identificados, iii) un buen funcionamiento dentro de la muestra no implica un buen ajuste de pronóstico fuera de la muestra.

2.6 INTERPRETACIÓN ECONÓMICA DE LAS SUPERFICIES ESCONDIDAS

La relación entre entradas (input) y salida (output) puede no ser directa En la literatura económica se encuentra evidencia de la existencia de variables intermedias o latentes que se ven afectadas por la variables inputs antes que el output[12]. En la modelación de las relaciones entre variables económicas con base en RNA las superficies escondidas o componentes logísticos de la red se asocian con las variables intermedias o latentes. Estas superficies caracterizan la no linealidad de la relación entre input y output.

Las superficies escondidas realizan un procesamiento intermedio de la información (inputs) antes de que la misma impacte el resultado final de la red (output), esto es, los inputs y su ponderación son sujetos a una reponderación antes de afectar el output. Este proceso intermedio de la red es del tipo "caja negra" debido a la dificultad para identificar y conocer todas las variables que afectan las relaciones económicas. Los modelos tipo "caja negra" no requieren para pronosticar series económicas del conocimiento o de la identificación de todas las variables que operan entre el input y output de acuerdo con Moody, Levin y Rehfuss (1993) y Moody (1995).

[12] En la relación entre tasa de interés e inflación probablemente se puede argumentar la existencia de muchas variables intermedias en el proceso de transmisión desde cambios en la tasa de interés a cambios en inflación. La tasa de interés puede primero afectar el consumo durable y la inversión, producción, el GAP de producto y por último la inflación.

3. RESULTADOS

La información mensual utilizada en el documento corresponde a la tasa de variación interanual del Índice de Precios al Consumidor (IPC), de la base monetaria (BM) y del medio circulante (M1). El periodo de análisis abarca de enero de 1990 a enero del 2005.

Como periodo de entrenamiento se utiliza el lapso comprendido entre enero de 1990 y diciembre de 1999, así la información para el pronóstico fuera de la muestra abarca 36 observaciones (2000:01-2002:12). Se escogen estos dos periodos con el fin de realizar una comparación entre los resultados de la Red Neuronal y los descritos en Durán et al (2003), obtenidos con otras técnicas de estimación.

La evaluación de la capacidad de pronóstico dentro y fuera de la muestra de los modelos estimados por RNA se realiza por medio de algunas medidas de evaluación del error de pronóstico. Entre estas medidas se tienen: la raíz del error cuadrático medio (RECM), raíz del error porcentual cuadrático medio (REPCM), error absoluto medio (EAM) y error porcentual absoluto medio (EPAM). Estos estadísticos miden la desviación de los valores pronosticados de inflación respecto a los observados en el periodo de análisis. La primera y la tercera medida del error dependen de la escala de la variable dependiente, por lo que deben ser utilizadas como medidas relativas para comparar pronósticos de la misma serie a través de diferentes modelos. Las otras dos medidas no dependen de la escala de la variable dependiente.

Los modelos de RNA que se ensayan fueron desarrollados en RATS, asumiendo una superficie escondida[13], de uno a cuatro nodos (Q) y un coeficiente de determinación (R^2) de 0,95 como criterio de convergencia.

Siguiendo la línea de investigación sobre el tema de la inflación implementada en Durán y otros (2003), en relación con los pronósticos de inflación, se parte de un modelo autorregresivo de inflación simple (modelo base) y se prueba si la incorporación de agregados monetarios (base monetaria y medio circulante) mejoran la precisión del pronóstico (modelos alternativos).

Para la elección del modelo base y de los alternativos se utiliza la estrategia denominada "paso a paso" (Stepwise), el cual es aplicado en Misas et al (2002) y es una propuesta de Swanson y White (1995, 1997a). Con esta técnica se parte de una especificación cuya variable dependiente es la tasa interanual de la inflación (aproximada por el IPC) y cuyas variables explicativas varían de acuerdo al modelo seleccionado. La elección final de la mejor red depende de las diferentes medidas de evaluación de la bondad de ajuste dentro y fuera de la muestra.

Un primer modelo (modelo base) incluye como variables explicativas los primeros cuatro rezagos de la inflación[14]. El segundo, y el tercero asumen como variables explicativas los rezagos de la inflación (del 1 al 4) junto con los rezagos de base

[13] RATS no cubre más de una superficie escondida, otros software como WinNN superan esta limitación.

[14] El número de rezagos del componente autorregresivo de la inflación se toma del trabajo Durán et al (2003).

monetaria (BM) y del medio circulante (M1), denominados modelos alternativos 1 y 2 respectivamente. Los agregados monetarios entran con un rezago de 24 periodos, esto es, con dos años de rezago[15]. El cuadro 1 resume ésta información.

Cuadro 1
Resultados Estrategia "paso a paso"

	Periodo de entrenamiento 1990-1999
Modelo base	IPC{1 a 4}
Modelo alternativo 1	IPC{1 a 4} BM{1 a 24}
Modelo alternativo 2	IPC{1 a 4} M1{1 a 24}

En el cuadro 2 se presentan los resultados de las diferentes medidas de evaluación dentro y fuera de la muestra para el modelo base y los modelos alternativos. Dentro de la muestra el modelo alternativo 2 se constituye como el mejor modelo de redes neuronales para el periodo comprendido entre 1990:01 y 1999:12 de acuerdo con las medidas de evaluación cuantificadas. Este modelo considera como insumos los rezagos del 1 al 4 de la inflación interanual, así como los rezagos del 1 al 24 del medio circulante, también medido como variación interanual. A la vez este modelo resulta ser el segundo mejor fuera de la muestra, superado en este caso por el modelo base.

En las figuras 4 a 6 y 7 a 9, que se presentan en el anexo, se muestra la inflación observada y la pronosticada por medio del modelo base y de los alternativos. En el primer conjunto de figuras, la inflación pronosticada corresponde al periodo de entrenamiento 19990:01-1999:12. En el segundo grupo de figuras se muestra el pronóstico fuera de muestra (2000:01-2002:12).

Cuadro 2
Modelos y Medidas de Evaluación

Dentro de Muestra (1990-1999)				Fuera de Muestra (2000-2002)			
RECM	REPCM	EAM	EPAM	RECM	REPCM	EAM	EPAM
Modelo base, Q=4							
1.45	8.88	1.09	6.60	0.86	8.63	0.68	6.61
Modelo alternativo 1, Q=4							
1.19	8.53	0.94	6.75	1.24	11.27	0.94	8.86
Modelo alternativo 2, Q=4							
1.19	8.19	0.92	6.32	1.05	10.95	0.79	7.93

[15] Para los agregados monetarios (base monetaria y medio circulante) se ensayaron otros rezagos sin resultados satisfactorios.

Como se mencionó anteriormente el periodo de entrenamiento para las redes neuronales, así como el de proyección fuera de muestra se fijó con el objetivo de realizar una comparación similar a la desarrollada en Durán et al (2003). En ese documento se evalúa la capacidad de proyección de los modelos que se han venido utilizando en la proyección combinada de inflación y los nuevos modelos seleccionados en la mencionada investigación.

El cuadro 3 detalla el desempeño de la red comparada con otros modelos que se actualizan mensualmente en la División Económica. Tomando como referencia la RECM sólo el Modelo Ingenuo supera a los identificados por la Red Neuronal Artificial para el periodo fuera de muestra comprendido entre enero de 2000 y diciembre de 2002.

Cuadro 3
Desempeño de la Red Versus Otros Modelos
2000-2002

	RECM
Red Neuronal Artificial	
Modelo base, Q=4	0.86
Modelo alternativo 1, Q=4	1.24
Modelo alternativo 2, Q=4	1.05
Modelos actuales del BCCR	
Modelo Ingenuo	0.71
Modelo de Precios del Petróleo	1.83
Pass Through	1.85
Modelo de Mecanismos	1.88
Modelo Títulos de Propiedad	2.02
Modelo ARMA	3.22

Para complementar los ejercicios anteriores las redes neuronales se entrenaron en el periodo 1990:01-2003:12 y se cuantificaron las medidas de error de pronóstico para el lapso 2004:01-2005:01. Los resultados se resumen en el cuadro 4.

Cuadro 4
Modelos y Medidas de Evaluación
(periodo de entrenamiento 1990-2003)

Fuera de Muestra (2004:01-2005:01)			
RECM	REPCM	EAM	EPAM
Modelo base, Q=4			
0.70	5.72	0.54	4.38
Modelo alternativo 1, Q=2			
1.05	8.40	0.91	7.32
Modelo alternativo 2, Q=4			
1.21	9.70	1.15	9.26

El considerar un segundo periodo de entrenamiento altera parcialmente los resultados que se detallan en el cuadro 2. El mejor ajuste de nuevo se obtiene con el modelo base, pero en este caso el modelo alternativo 1 supera al alternativo 2. En el anexo se incluyen las figuras 10-12 que muestran la inflación observada y pronostica por cada

modelo neuronal. El modelo base y el alternativo 1 hacia el final del horizonte de proyección ajustan bastante bien, en el caso del modelo base llama la atención que el excelente ajuste en el último trimestre del 2004.

La proyección de inflación mensual[16] para el periodo setiembre-diciembre de 2005 de los modelos de redes neuronales se muestra en el cuadro 5.

Cuadro 5
RNA: Proyección de inflación
(periodo de entrenamiento 1990:01- 2005:08)

Un paso fuera de Muestra (2005:09-2005:12)			
Setiembre	Octubre	Noviembre	Diciembre
Modelo base, Q=4			
13.7	13.6	13.4	13.3
Modelo alternativo 1, Q=2			
12.9	13.7	13.7	12.9
Modelo alternativo 2, Q=4			
12.2	13.0	13.8	13.2

Nota: Se cuantificó la medida de error de pronóstico (RECM) para el lapso 2005:01-2005:08, estos fueron 0,49, 0,72 y 0,58 para el modelo base, alternativo 1 y alternartivo 2, respectivamente.

4. CONCLUSIONES

El desempeño de los modelos estimados con la técnica de Redes Neuronales Artificiales, dentro de la muestra, evidencian una mejora en la precisión del pronóstico de inflación cuando se incorporan agregados monetarios. Así, el modelo alternativo 2 se selecciona como el de mejor ajuste.

En los dos periodos evaluados fuera de la muestra con esta técnica el modelo base supera a los alternativos. No obstante lo anterior, los modelos alternativos, que incorporan agregados monetarios, muestran un buen desempeño en la proyección de inflación de acuerdo con las medidas de error de pronóstico.

[16] La proyección se realiza un paso fuera de la muestra a la vez, en el caso de las variables monetarias (Medio Circulante y Base Monetaria) se utilizan los saldos mensuales proyectados en la revisión del Programa Monetaria 2005-2006.

En relación con los modelos que ha venido actualizando la División Económica[17] los estimados por medio de la RNA se pueden considerar competitivos. Sólo el Modelo Ingenuo no es superado.

Con estos resultados se amplía el grupo de modelos disponibles para el pronóstico de la inflación. Al menos dos de los modelos incluyen como insumo variables monetarias, ausentes en los otros modelos.

La técnica de RNA es una herramienta con un alto potencial en el pronóstico y modelación de variables dado que permite relaciones univariables o multivariables no lineales muy generales, las cuales podrían ser difíciles de ajustar usando otras técnicas.

17 Los modelos a que se hacen referencia son: El modelo ARMA corresponde a un ARMA(6,3) (Hoffmaister et al., 2000a). El modelo VAR incorpora la tasa de interés internacional (Estados Unidos), e incluye además los precios domésticos, el tipo de cambio, la tasa de interés doméstica, la actividad económica, y crédito total al sector privado (Flores et al., 2000). El modelo de Petróleo consiste en un modelo VAR que hace explícita la regla que gobierna los precios domésticos de los combustibles (Hoffmaister et al, 2000b). El modelo ingenuo proyecta la inflación asumiendo que no varía con respecto al último dato observado. El modelo Pass through incluye inflación rezagada, devaluación, desviación del tipo de cambio real respecto a su tendencia, gap de la desviación del producto respecto a su tendencia y un indicador de apertura (León et al, 2001). El modelo TP incluye variación del saldo colocado de bonos del gobierno central, inflación rezagada y la tasa básica pasiva (Duran et al, 2003).

5. REFERENCIAS

Cover, J., "Asymmetric Effects of Positive and Negative Money-Supply Shocks", Quarterly Journal of Economics 107: 1260-1282, 1992.

Durán V., Rodolfo, Bernal Laverde y Mario G. Solano, "Evaluación de Variables Económicas para Pronósticos de Inflación en Costa Rica", DIE-02-2003-DI, Documento de Investigación, Marzo, 2003.

Evans, O.V:D., "Short-Term Currency Forecasting Using Neural Networks", ICL Systems Journal, 1997.

Flores, Melania, Alexander W. Hoffmaister, Jorge Madrigal, y Lorely Villalobos, Transmisión Monetaria en Costa Rica, Nota de Investigación No. 3-00, Banco Central de Costa Rica, Agosto 2000.

Franses.P., and D: van Dijk, "Non-linear time series models in empirical finance, Cambridge University Press, 2000.

Friedman, M., "The Role of Monetary Policy", American Economic Review 58, 1968.

Guía del Usuario, RATS versión 5, Modelos Especiales, Capítulo 12.

Herbrich, R., M. Keilbach, Graepel, T., bollmann, P. And K. Obermayer, "Neural Networks in Economics: Background, Application and New Developments", Technische universitat Berlin, 1999.

Hoffmaister, Alexander W., Ivannia Solano, Álvaro Solera, y Katia Vindas, Modelos Univariables de la Inflación, Nota de Investigación No. 5-00, Banco Central de Costa Rica, agosto 2000b.

Hoffmaister, Alexander W., Ivannia Solano, Álvaro Solera, y Katia Vindas, Impacto de los Precios del Petróleo en Costa Rica, Nota de Investigación No. 4-00, Banco Central de Costa Rica, setiembre 2000c.

Kuan, C.M. and H. White,"Artificial Neural Networks: An Econometric Perspective", Econometric Review 13, 1994.

León M., Jorge, Ana P. Morera M. y Welmer Ramos Gónzalez, "El Pass Through del Tipo de Cambio: Un Análisis de la Economía Costarricense de 1991 al 2001", *DIE-DM/11-2001-DI*, División Económica, Banco Central de Costa Rica, Diciembre, 2001.

Mayorga M., Mauricio, Juan Carlos Quirós y Alvaro Solera, "Efectos Asimétricos de la Política Monetaria", DIE-01-2003-DI, Documento de Investigación, Febrero, 2003.

Misas Arango, Martha, Enrique López Enciso y Pablo Querubín Borrero, "La Inflación en Colombia: Una Aproximación desde las Redes Neuronales", Subgerencia de Estudios Económicos, Banco de la República, febrero, 2002.

Moody, J.,"Economic Forecasting: Challenges and Neural Network solutions", International Symposium on Artificial Neural Networks, Taiwan, 1995.

Moody, J., U. Levin and S. Rehfuss, "Predicting the U.S. Index of Industrial Production", Appears in Proceedings of Parallel Application in Statistic And Economics, Special Issue of Neural Network World, Vol. 3, No 6, 1993.

Morgan, D.P., "Asymmetric Effects of Monetary Policy", Federal Reserve Bank of Kansas City, Econometric Review 78:21-33.

Moshiri, S., and N. Cameron, "Neural Networks vs Econometric Models in Forecasting Inflation", Departament of Economics, University of Manitoba, 1998.

Serju, Prudence, "Monetary Conditions and Core Inflation: An Application of Neural Networks", Research Services Departament, Bank of Jamaica, 2000.

Swanson, N.R. and H. White, "A Model-Selection Approach to Assessing the Information in the Term Structure Using Linear Models and Artificial Neural Networks", Journal of Business & Economic Statistics, Vol 13, No 3, 1995.

Swanson, N.R. and H. White, "A Model-Selection Approach to Real-Time Macroeconomic Forecasting Using Linear Models and Artificial Neural Networks", The Review of Economics and Statistics, No 79, 1977a.

Tkacz, Greg, "Non-Parametric and Neural Network Models of Inflation Changes", Working Paper 2000-7, Bank of Canada, April, 2000.

Tkacz, Greg and S. Hu, "Forecasting GDP Growth Using Artificial Neural Networks", Working Paper 99-3, Bank of Canada, 1999.

solerara@bccr.fi.cr

6. ANEXOS

6.1 FIGURAS

Figura 4. Pronóstico del IPC con la RNA (dentro de la muestra), 1990:01-1999:12

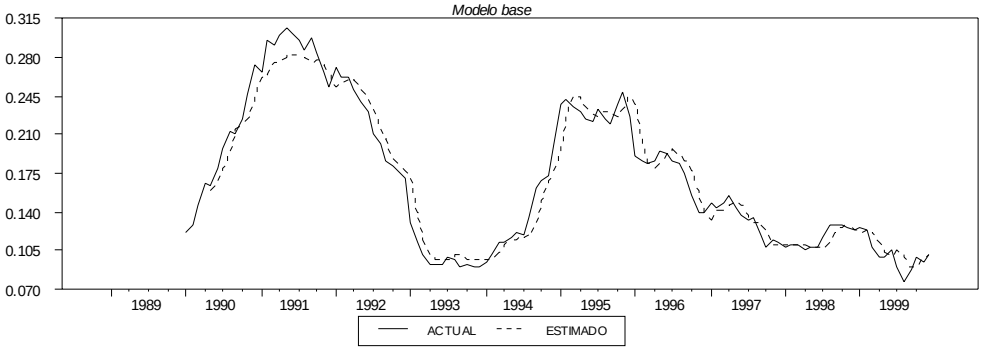


Figura 5. Pronóstico del IPC con la RNA (dentro de la muestra), 1990:01-1999:12

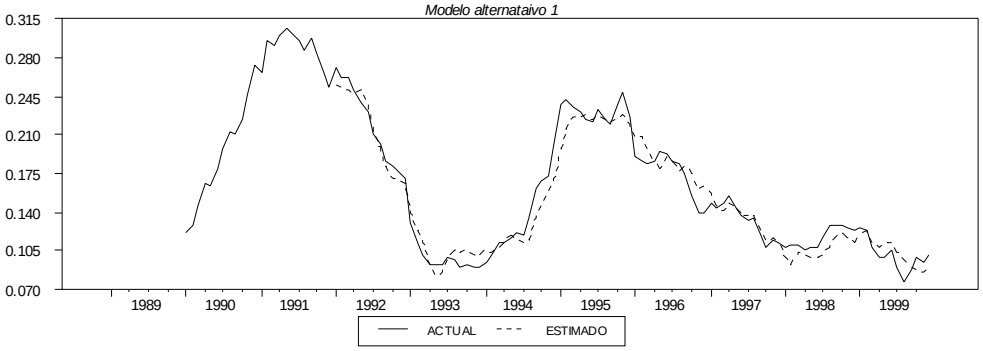


Figura 6. Pronóstico del IPC con la RNA (dentro de la muestra), 1990:01-1999:12

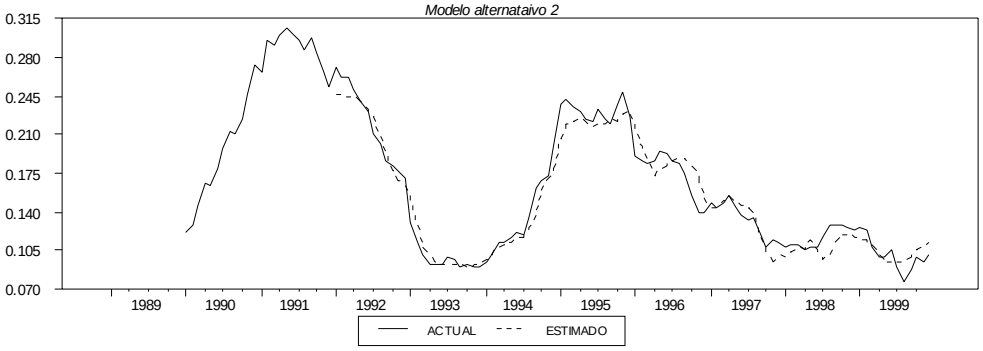


Figura 7. Pronóstico del IPC con la RNA (fuera de la muestra), 2000:01-2002:12

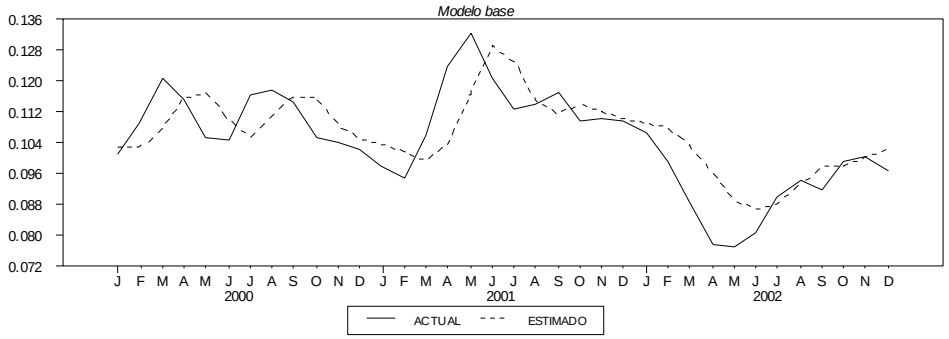


Figura 8. Pronóstico del IPC con la RNA (fuera de la muestra), 2000-2002

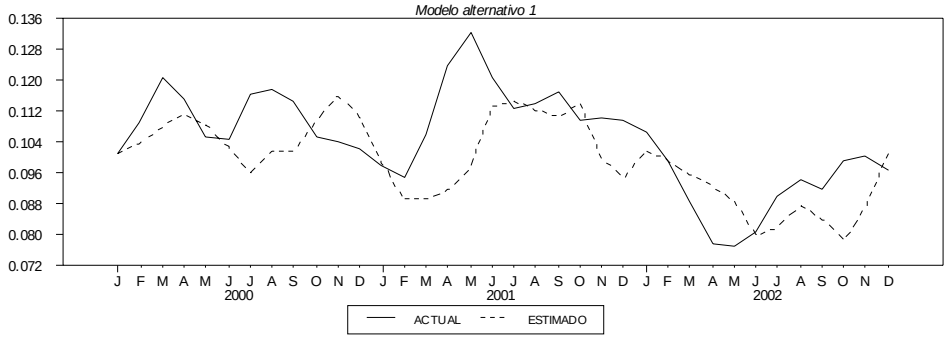


Figura 9. Pronóstico del IPC con la RNA (fuera de la muestra), 2000-2002

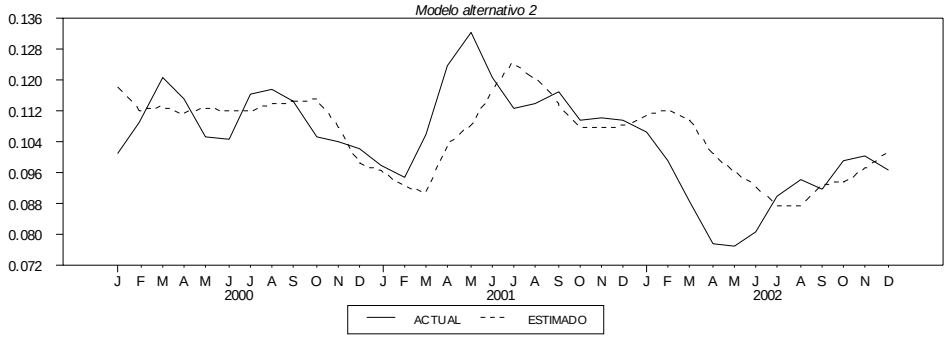


Figura 10. Pronóstico del IPC con la RNA (fuera de la muestra), 2004:01-2005:01

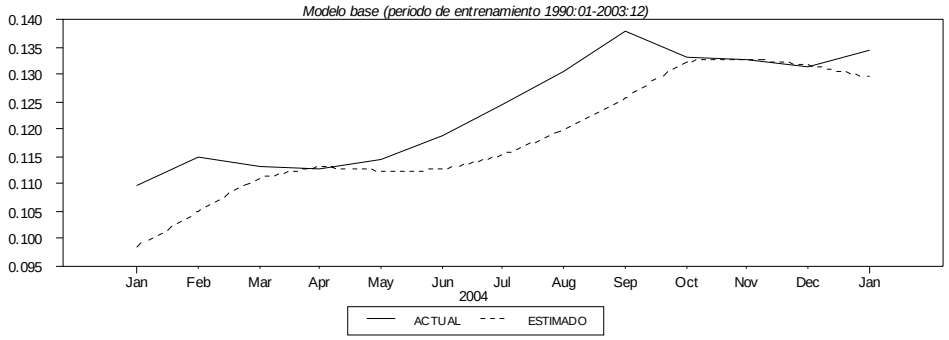


Figura 11. Pronóstico del IPC con la RNA (fuera de la muestra), 2004:01-2005:01

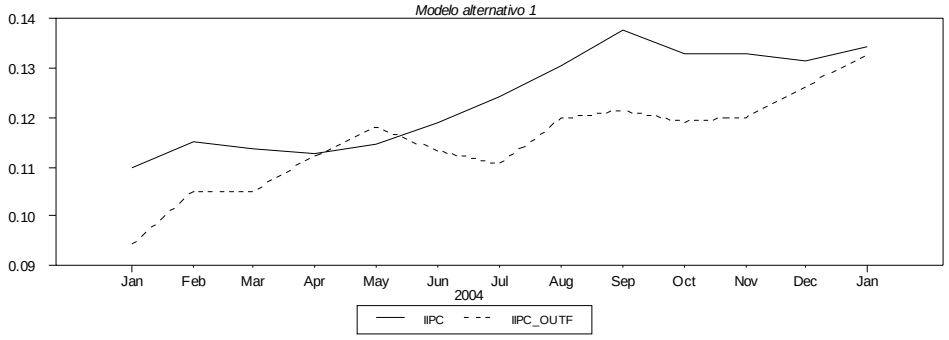
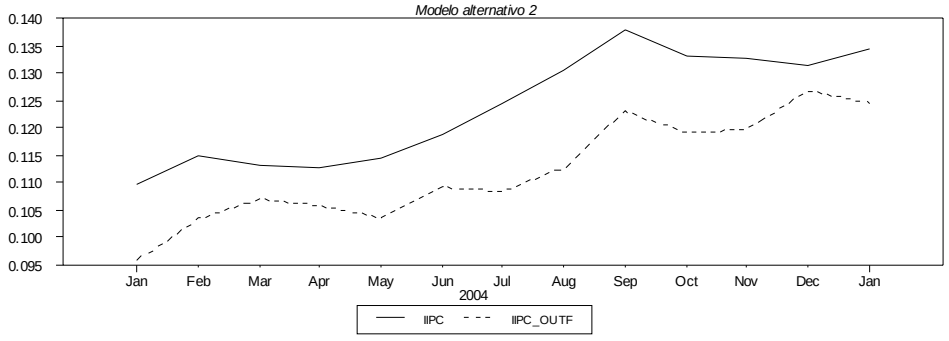


Figura 12. Pronóstico del IPC con la RNA (fuera de la muestra), 2004:01-2005:01



6.2 PROGRAMACIÓN EN RATS PARA LA ESTIMACIÓN DE REDES NEURONALES ARTIFICIALES.

```
*************************************************************************
*                                                                       *
*                    PROGRAMA PARA                                      *
*              REDES NEURONALES ARTIFICIALES                            *
*************************************************************************
*                      23/03/2005
*-----------------------------------------------------------------------
CALENDAR 1989 1 12
ALLOCATE 0 2010:12
*
*LECTURA DE DATOS
*
OPEN DATA P:\ANN\baseann1.xls
DATA(FORMAT=XLS,ORG=OBS) / IPC M1 M2 M3 BM BEM6 IMAE TC CPRIV BRENT
TBILL WTI
*PRINT / IPC M1 M2 M3 BM BEM6 IMAE TC CPRIV BRENT TBILL WTI

*HALT
*------------------------------parámetros-------------------------------------
*
COMPUTE START    = 1989:01
COMPUTE START1   = 2000:01
*
COMPUTE END      = 2002:12
compute end1     = 1999:12
*
COMPUTE START_F  = 2003:01
COMPUTE END_F    = 2005:01
*
COMPUTE START_F1 = 2000:01
COMPUTE END_F1   = 2002:12
*
COMPUTE START_F2 = 2004:01
COMPUTE END_F2   = 2005:01
*
COMPUTE P        = 4    ;* NÚMERO DE REZAGOS
*
COMPUTE HORIZON  = 25   ;* HORIZONTE DE PROYECCION
*
*-------------------------Cálculo tasas interanuales----------------------
*
SET IIPC             / = (IPC/IPC{12}-1.0)
SET ITC                  / = (TC/TC{12}-1.0)
SET IIMAE            / = (IMAE/IMAE{12}-1.0)
SET ICPRIV           / = (CPRIV/CPRIV{12}-1.0)
SET IBRENT           / = (BRENT/BRENT{12}-1.0)
SET IWTI             / = (WTI/WTI{12}-1.0)
SET ITBILL           / = (TBILL/TBILL{12}-1.0)
SET IBEM6            / = (BEM6/100)
SET IM1            / = (M1/M1{12}-1.0)
SET IM2              / = (M2/M2{12}-1.0)
SET IM3            / = (M3/M3{12}-1.0)
SET IBM             / = (BM/BM{12}-1.0)
*
```

```
set iipc_out        / = 0.0
set iipc_outf       / = 0.0
set iipc_out1       / = 0.0

*PRINT / IIPC IBM IM1 ITC IIMAE IBRENT IBEM6
*HALT
*
*********************************************************
*------------MODELO BASE-----------------------------
*********************************************************
*MODELACIÓN DE LA RED NEURONAL ARTIFICIAL
*NNLEARN define un nuevo modelo RN y realiza entrenamientos
*adicionales para un modelo existente.Se utiliza semilla.
*
DECLARE RECTANGULAR ERRORESMOD1D(6,4)
DECLARE VECTOR[SERIES] ERROR(6) ERR0R2(6) error3(6) error4(6)
*
seed 10
*Modelo 1: rezagos del IPC(4) vs IPC, considerando de 1 a 4 unidades (nodos)
*escondidas en una superficie escondida dentro de la muestra
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV,TRACE,MODE=EPOCH) start end1
  # IIPC{1 TO P}
  # IIPC
  *
  compute array = MV
  *
  *NNTEST genera la salida (valores ajustados) desde el conjunto de
  *inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST start end1 array ;*ajuste dentro de la muestra
  # IIPC{1 TO P}
  # iipc_out
  *
  set actual start end1 = IIPC
  set estimado start end1 = iipc_out
  *
  PRINT start end1 IIPC iipc_out actual estimado
  *halt
  OPEN PLOT IPCDM.RGF
  *
  graph(key=below,pattern,$
  HEADER='Figura 4. Pronóstico del IPC con la RNA (dentro de la muestra)1990:01-1999:12',$
  subheader ='Modelo base') 2
  # actual     start end1
  # estimado   start end1
  *halt
  close plot
  *
  *Calculo de la raíz del error cuadrático medio (RECM)
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = iipc_out - iipc
  set ERR0R2(j)       start end1 = error(j) * error(j)
  accumalate ERR0R2(j) start end1   error3(j)
  set ERROR4(j)       end1  end1  = (ERROR3(j)/(120-p))**0.5
  *
  make temp1 end1 end1
```

```
  # error4(j)
  *
  write temp1
  *
  compute erroresmod1d(j,1) = temp1(1,1)
  *
  write erroresmod1d
  *
  *Calculo de la raíz del error porcentual cuadrático medio (REPCM)
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = ((iipc_out - iipc)/iipc)
  set ERR0R2(j)       start end1 = error(j) * error(j)
  accumalate ERR0R2(j) start end1 error3(j)
  set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p))**0.5
  *
  make temp1 end1 end1
  # error4(j)
  *
  write temp1
  *
  compute erroresmod1d(j,2) = temp1(1,1)
  *
  write erroresmod1d
  *
  *Calculo del error absoluto medio
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = abs(iipc_out - iipc)
  set ERR0R2(j)       start end1 = error(j) * 1
  accumalate ERR0R2(j) start end1 error3(j)
  set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p))
  *
  make temp1 end1 end1
  # error4(j)
  *
  write temp1
  *
  compute erroresmod1d(j,3) = temp1(1,1)
  *
  write erroresmod1d
  *
  *Calculo del error porcentual absoluto medio
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = abs((iipc_out - iipc)/iipc)
  set ERR0R2(j)       start end1 = error(j) * 1
  accumalate ERR0R2(j) start end1 error3(j)
  set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p))
  *
  make temp1 end1 end1
  # error4(j)
  *
  write temp1
  *
  compute erroresmod1d(j,4) = temp1(1,1)
  *
  write erroresmod1d
*
end do j
```

```
*
open copy erroresmod1d.dat
write(unit=copy) erroresmod1d
close copy
*
*halt
*
*
* Estadísticos de bondad de ajuste para el periodo fuera de muestra ******
* 2004:01-2005:01* periodo de entrenamiento 1990:01-2003:12 **************
*************************************************************************
DECLARE RECTANGULAR ERRORES(4,4)
DECLARE VECTOR[SERIES] ERROR(4) ERR0R2(4) error3(4) error4(4)
*
seed 10
*Modelo 1: rezagos del IPC(4) vs IPC, considerando de 1 a 4 unidades (nodos)
*escondidas en una superficie escondida dentro de la muestra
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV,TRACE,MODE=EPOCH) start 2003:12
  # IIPC{1 TO P}
  # IIPC
  *
  compute array = MV
  *
  *NNTEST genera la salida (valores ajustados) desde el conjunto de
  *inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST start_f2 end_f2 array ;*ajuste fura de la muestra
  # IIPC{1 TO P}
  # iipc_out1
  *
  set actual   start_f2 end_f2 = IIPC
  set estimado start_f2 end_f2 = iipc_out1
  *
  PRINT start_f2 end_f2 IIPC iipc_out1  actual estimado
  *halt
  *
  open plot IPCFM04-05.rgf
  graph(key=below,pattern,HEADER='Figura 10. Pronóstico del IPC con la RNA (fuera de la
muestra), 2004:01-2005:01',$
  subheader ='Modelo base (periodo de entrenamiento 1990:01-2003:12)') 2
  # actual     start_f2 end_f2
  # estimado    start_f2 end_f2
  close plot
  *halt
  *
  *Calculo de la raíz del error cuadrático medio (RECM)
  *fuera de la muestra 2004:01-2005:01
  *
  set error(j)        start_f2 end_f2 = iipc_out1 - iipc
  set ERR0R2(j)       start_f2 end_f2 = error(j) * error(j)
  accumalate ERR0R2(j) start_f2 end_f2   error3(j)
  set ERROR4(j)       end_f2  end_f2  = (ERROR3(j)/(13))**0.5
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
```

```
  *
  compute errores(j,1) = temp1(1,1)
  *
  write errores
  *
  *Calculo de la raíz del error porcentual cuadrático medio (REPCM)
  *fuera de la muestra 2004:01-2005:01
  *
  set error(j)        start_f2 end_f2 = ((iipc_out1 - iipc)/iipc)
  set ERR0R2(j)       start_f2 end_f2 = error(j) * error(j)
  accumalate ERR0R2(j) start_f2 end_f2  error3(j)
  set ERROR4(j)       end_f2  end_f2 = (ERROR3(j)/(13))**0.5
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
  *
  compute errores(j,2) = temp1(1,1)
  *
  write errores
  *
  *Calculo del error absoluto medio
  *fuera de la muestra 2004:01-2005:01
  *
  set error(j)        start_f2 end_f2 = abs(iipc_out1 - iipc)
  set ERR0R2(j)       start_f2 end_f2 = error(j) * 1
  accumalate ERR0R2(j) start_f2 end_f2  error3(j)
  set ERROR4(j)       end_f2  end_f2 = (ERROR3(j)/(13))
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
  *
  compute errores(j,3) = temp1(1,1)
  *
  write errores
  *
  *Calculo del error porcentual absoluto medio
  *fuera de la muestra 2004:01-2005:01
  *
  set error(j)        start_f2 end_f2 = abs((iipc_out1 - iipc)/iipc)
  set ERR0R2(j)       start_f2 end_f2 = error(j) * 1
  accumalate ERR0R2(j) start_f2 end_f2  error3(j)
  set ERROR4(j)       end_f2  end_f2 = (ERROR3(j)/(13))
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
  *
  compute errores(j,4) = temp1(1,1)
  *
  write errores
*
end do j
*
open copy errores.dat
write(unit=copy) errores
```

```
close copy
*
*halt
*
*************************************************************************
seed 10
*Modelo 1: rezagos del IPC(4) vs IPC, considerando de 1 a 4 unidades (nodos)
*escondidas en una superficie escondida dentro de la muestra
*para realizar pronóstico a diciembre 2005 un paso fuera de muestra
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV,TRACE,MODE=EPOCH) start 2005:08
  # IIPC{1 TO P}
  # IIPC
  *
  compute array = MV
  *
  *NNTEST genera la salida (valores ajustados) desde el conjunto de
  *inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST 2005:01 2005:12 array ;*ajuste fuera de la muestra
  # IIPC{1 TO P}
  # iipc_out1
  *
  set actual   2005:01 2005:12 = IIPC
  set estimado 2005:01 2005:12 = iipc_out1
  *
  PRINT 2005:01 2005:12 IIPC iipc_out1  actual estimado

end do j
*halt
*********************************************************************
*
*---------estimación fuera de la muestra:2000:01-2002:12--------
****periodo de entrenamiento 1990:01-1999:12*******************
*********************************************************************
*********************************************************************
DECLARE RECTANGULAR ERRORESMOD1F(4,4)
DECLARE VECTOR[SERIES] ERROR(4) ERR0R2(4) error3(4) error4(4)
*
*Modelo 1: rezagos del IPC(4) vs IPC, considerando de 1 a 4 unidades
*escondidas en una superficie escondida fuera de la muestra
*
seed 10
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV,TRACE,MODE=EPOCH) start end1
  # IIPC{1 TO P}
  # IIPC
  *
  compute array = MV
  *
  *NNTEST genera la salida (valores ajustados fuera de la muestra)
  *desde el conjunto de*inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST start_f1 end_f1 array ;*ajuste fuera de la muestra
  # IIPC{1 TO P}
  # iipc_outf
  *
```

```
set actual start_f1 end_f1   = IIPC
set estimado start_f1 end_f1 = iipc_outf
*
PRINT start_f1 end_f1 iipc iipc_outf actual estimado
open plot IPCFM00-02.rgf
graph(key=below,pattern,HEADER='Figura 7. Pronóstico del IPC con la RNA (fuera de la
muestra), 2000:01-2002:12',$
subheader ='Modelo base') 2
# actual      start_f1 end_f1
# estimado    start_f1 end_f1
close plot
*halt
*
*Calculo de la raíz del error cuadrático medio (RECM)
*fuera de la muestra 2000:01-2002:12
*
set error(j)        start_f1 end_f1 = iipc_outf - iipc
set ERR0R2(j)       start_f1 end_f1 = error(j) * error(j)
accumalate ERR0R2(j) start_f1 end_f1  error3(j)
set ERROR4(j)       end_f1 end_f1  = (ERROR3(j)/(36))**0.5
*
make temp1 end_f1 end_f1
# error4(j)
*
write temp1
*
compute erroresmod1f(j,1) = temp1(1,1)
*
write erroresmod1f
*
*Calculo de la raíz del error porcentual cuadrático medio (REPCM)
*fuera de la muestra 2000:01-2002:12
*
*
set error(j)        start_f1 end_f1 = ((iipc_outf - iipc)/iipc)
set ERR0R2(j)       start_f1 end_f1 = error(j) * error(j)
accumalate ERR0R2(j) start_f1 end_f1 error3(j)
set ERROR4(j)       end_f1 end_f1 = (ERROR3(j)/(36))**0.5
*
make temp1 end_f1 end_f1
# error4(j)
*
write temp1
*
compute erroresmod1f(j,2) = temp1(1,1)
*
write erroresmod1f
*
*Calculo ded error absoluto medio (MAE)
*fuera de la muestra 2000:01-2002:12
*
set error(j)        start_f1 end_f1 = abs(iipc_outf - iipc)
set ERR0R2(j)       start_f1 end_f1 = error(j) * 1
accumalate ERR0R2(j) start_f1 end_f1 error3(j)
set ERROR4(j)       end_f1 end_f1 = (ERROR3(j)/(36))
*
make temp1 end_f1 end_f1
# error4(j)
*
write temp1
```

```
  *
  compute erroresmod1f(j,3) = temp1(1,1)
  *
  write erroresmod1f
  *
  *Calculo del error porcentual medio absoluto (MAPE)
  *fuera de la muestra 2000:01-2002:12
  *
  set error(j)        start_f1 end_f1 = abs((iipc_outf - iipc)/iipc)
  set ERR0R2(j)       start_f1 end_f1 = error(j) * 1
  accumalate ERR0R2(j) start_f1 end_f1 error3(j)
  set ERROR4(j)       end_f1 end_f1 = (ERROR3(j)/(36))
  *
  make temp1 end_f1 end_f1
  # error4(j)
  *
  write temp1
  *
  compute erroresmod1f(j,4) = temp1(1,1)
  *
  write erroresmod1f
*
end do j
*
open copy erroresmod1f.dat
write(unit=copy) erroresmod1f
close copy
*
*end
*halt
*
**********************************************************************
*-------------MODELO ALTERNATIVO 1 (se incluye BM)----------------
**********************************************************************
*
*MODELACIÓN DE LA RED NEURONAL ARTIFICIAL
*NNLEARN define un nuevo modelo RN y realiza entrenamientos
*adicionales para un modelo existente.Se utiliza semilla.
*
DECLARE RECTANGULAR ERRORESBM(4,4)
DECLARE VECTOR[SERIES] ERROR(4) ERR0R2(4) error3(4) error4(4)
*
COMPUTE P1       = 24
COMPUTE P2       = 9
COMPUTE P3       = 3
seed 10
*
*Modelo 2: rezagos del IPC(4),BM(24) vs IPC, considerando de 1 a 4 unidades (nodos)
*escondidas en una superficie escondida dentro de la muestra
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV1,TRACE,MODE=EPOCH) start end1
  # IIPC{1 TO P} Ibm{1 TO p1};* ITC{1 TO p2}
  # IIPC
  *
  compute array = MV1
  *
  *NNTEST genera la salida (valores ajustados) desde el conjunto de
  *inputs, usando el modelo estimado con NNLEARN
```

```
*
NNTEST start end1 array ;*ajuste dentro de la muestra
# IIPC{1 TO P} Ibm{1 TO p1} ;*ITC{1 TO p2}
# iipc_out
*
set actual start end1   = IIPC
set estimado start end1 = iipc_out
*
PRINT start end1 IIPC iipc_out actual estimado
*halt
open plot IPC_BMDM.rgf
graph(key=below,pattern,HEADER='Figura 5. Pronóstico del IPC con la RNA (dentro de la
muestra), 1990:01-1999:12',$
subheader ='Modelo alternataivo 1') 2
# actual    start end1
# estimado  start end1
close plot
*halt
*
*Calculo de la raíz del error cuadrático medio (RECM)
*dentro de la muestra 1990:01-1999:12
*
set error(j)        start end1 = iipc_out - iipc
set ERR0R2(j)       start end1 = error(j) * error(j)
accumalate ERR0R2(j) start end1   error3(j)
set ERROR4(j)       end1  end1  = (ERROR3(j)/(120-p1))**0.5
*
*
make temp1 end1 end1
# error4(j)
*
write temp1
*
compute erroresBM(j,1) = temp1(1,1)
*
write erroresBM
*
*Calculo de la raíz del error porcentual cuadrático medio (REPCM)
*dentro de la muestra 1990:01-1999:12
*
set error(j)        start end1 = ((iipc_out - iipc)/iipc)
set ERR0R2(j)       start end1 = error(j) * error(j)
accumalate ERR0R2(j) start end1 error3(j)
set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p1))**0.5
*
make temp1 end1 end1
# error4(j)
*
write temp1
*
compute erroresBM(j,2) = temp1(1,1)
*
write erroresBM
*
*Calculo del error absoluto medio
*dentro de la muestra 1990:01-1999:12
*
set error(j)        start end1 = abs(iipc_out - iipc)
set ERR0R2(j)       start end1 = error(j) * 1
accumalate ERR0R2(j) start end1 error3(j)
```

```
  set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p1))
  *
  make temp1 end1 end1
  # error4(j)
  *
  write temp1
  *
  compute erroresBM(j,3) = temp1(1,1)
  *
  write erroresBM
  *
  *Calculo del error porcentual absoluto medio
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = abs((iipc_out - iipc)/iipc)
  set ERR0R2(j)       start end1 = error(j) * 1
  accumalate ERR0R2(j) start end1 error3(j)
  set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p1))
  *
  make temp1 end1 end1
  # error4(j)
  *
  write temp1
  *
  compute erroresBM(j,4) = temp1(1,1)
  *
  write erroresBM
*
end do j
*
open copy erroresBM.dat
write(unit=copy) erroresBM
close copy
*
*halt
*
* Estadísticos de bondad de ajuste para el periodo 2000-2002***
*****************************************************************
DECLARE RECTANGULAR ERRORESbm1(4,4)
DECLARE VECTOR[SERIES] ERROR(4) ERR0R2(4) error3(4) error4(4)
*
seed 10
*Modelo 2: rezagos del IPC(4),rezagos BM vs IPC, considerando de 1 a 4 unidades (nodos)
*escondidas en una superficie escondida dentro de la muestra
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV1,TRACE,MODE=EPOCH) start end1
  # IIPC{1 TO P} Ibm{1 TO P1}
  # IIPC
  *
  compute array = MV1
  *
  *NNTEST genera la salida (valores ajustados) desde el conjunto de
  *inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST start_f1 end_f1 array ;*
  # IIPC{1 TO P} Ibm{1 TO P1}
  # iipc_out1
  *
```

```
set actual   start_f1 end_f1 = iipc
set estimado start_f1 end_f1 = iipc_out1
*
PRINT start_f1 end_f1 IIPC iipc_out1 actual estimado
*halt
open plot IPC_BMFM00-02.rgf
graph(key=below,pattern,HEADER='Figura 8. Pronóstico del IPC con la RNA (fuera de la
muestra), 2000-2002',$
subheader ='Modelo alternativo 1') 2
# actual      start_f1 end_f1
# estimado    start_f1 end_f1
close plot
*halt
*
*Calculo de la raíz del error cuadrático medio (RECM)
*fuera de la muestra 2000-2002
*
set error(j)        start_f1 end_f1 = iipc_out1 - iipc
set ERR0R2(j)       start_f1 end_f1 = error(j) * error(j)
accumalate ERR0R2(j) start_f1 end_f1   error3(j)
set ERROR4(j)       end_f1  end_f1  = (ERROR3(j)/(36))**0.5
*
*
make temp1 end_f1 end_f1
# error4(j)
*
write temp1
*
compute erroresbm1(j,1) = temp1(1,1)
*
write erroresbm1
*
*Calculo de la raíz del error porcentual cuadrático medio (REPCM)
*fuera de la muestra 2000-2002
*
set error(j)        start_f1 end_f1 = ((iipc_out1 - iipc)/iipc)
set ERR0R2(j)       start_f1 end_f1 = error(j) * error(j)
accumalate ERR0R2(j) start_f1 end_f1 error3(j)
set ERROR4(j)       end_f1  end_f1 = (ERROR3(j)/(36))**0.5
*
make temp1 end_f1 end_f1
# error4(j)
*
write temp1
*
compute erroresbm1(j,2) = temp1(1,1)
*
write erroresbm1
*
*Calculo del error absoluto medio
*dentro de la muestra 2000-2002
*
set error(j)        start_f1 end_f1 = abs(iipc_out1 - iipc)
set ERR0R2(j)       start_f1 end_f1 = error(j) * 1
accumalate ERR0R2(j) start_f1 end_f1 error3(j)
set ERROR4(j)       end_f1  end_f1 = (ERROR3(j)/(36))
*
make temp1 end_f1 end_f1
# error4(j)
*
```

```
  write temp1
  *
  compute erroresbm1(j,3) = temp1(1,1)
  *
  write erroresbm1
  *
  *Calculo del error porcentual absoluto medio
  *fuera de la muestra 2000-2002
  *
  set error(j)        start_f1 end_f1 = abs((iipc_out1 - iipc)/iipc)
  set ERR0R2(j)       start_f1 end_f1 = error(j) * 1
  accumalate ERR0R2(j) start_f1 end_f1 error3(j)
  set ERROR4(j)       end_f1  end_f1 = (ERROR3(j)/(36))
  *
  make temp1 end_f1 end_f1
  # error4(j)
  *
  write temp1
  *
  compute erroresbm1(j,4) = temp1(1,1)
  *
  write erroresbm1
*
end do j
*
open copy erroresbm1.dat
write(unit=copy) erroresbm1
close copy
*
*halt
*
*---------estimación fuera de la muestraIPC_BM 2004:01-2005:01----------
*
DECLARE RECTANGULAR ERRORESbm2(4,4)
DECLARE VECTOR[SERIES] ERROR(4) ERR0R2(4) error3(4) error4(4)
*
*Modelo 2: rezagos del IPC(4), BM(24) vs IPC, considerando de 1 a 4 unidades
*escondidas en una superficie escondida fuera de la muestra
*
seed 10
*
Do j= 1,2
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV1,TRACE,MODE=EPOCH) start 2003:12
  # IIPC{1 TO P} Ibm{1 TO P1}
  # IIPC
  *
  compute array = MV1
  *
  *NNTEST genera la salida (valores ajustados fuera de la muestra)
  *desde el conjunto de*inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST start_f2 end_f2 array ;*ajuste fuera de la muestra
  # IIPC{1 TO P} Ibm{1 TO P1}
  # iipc_outf

  set actual   start_f2 end_f2 = iipc
  set estimado start_f2 end_f2 = iipc_outf
  *
  *
```

```
PRINT start_f2 end_f2 iipc iipc_outf actual estimado
open plot IPC_BMFM04-05.rgf
graph(key=below,pattern,HEADER='Figura 11. Pronóstico del IPC con la RNA (fuera de la
muestra), 2004:01-2005:01',$
subheader ='Modelo alternativo 1') 2
# iipc       start_f2 end_f2
# iipc_outf  start_f2 end_f2
close plot
*halt
*
*Calculo de la raíz del error cuadrático medio (RECM)
*fuera de la muestra 2004:01-2005:01
*
set error(j)        start_f2 end_f2 = iipc_outf - iipc
set ERR0R2(j)       start_f2 end_f2 = error(j) * error(j)
accumalate ERR0R2(j) start_f2 end_f2  error3(j)
set ERROR4(j)       start_f2 end_f2  = (ERROR3(j)/(13))**0.5
*
*
make temp1 end_f2 end_f2
# error4(j)
*
write temp1
*
compute erroresbm2(j,1) = temp1(1,1)
*
write erroresbm2
*
*Calculo de la raíz del error porcentual cuadrático medio (REPCM)
*fuera de la muestra 2004:01-2005:01
*
set error(j)        start_f2 end_f2 = ((iipc_outf - iipc)/iipc)
set ERR0R2(j)       start_f2 end_f2 = error(j) * error(j)
accumalate ERR0R2(j) start_f2 end_f2 error3(j)
set ERROR4(j)       start_f2 end_f2 = (ERROR3(j)/(13))**0.5
*
make temp1 end_f2 end_f2
# error4(j)
*
write temp1
*
compute erroresbm2(j,2) = temp1(1,1)
*
write erroresbm2
*
*Calculo ded error absoluto medio (MAE)
*fuera de la muestra 2004:01-2005:01

*
set error(j)        start_f2 end_f2 = abs(iipc_outf - iipc)
set ERR0R2(j)       start_f2 end_f2 = error(j) * 1
accumalate ERR0R2(j) start_f2 end_f2 error3(j)
set ERROR4(j)       start_f2 end_f2 = (ERROR3(j)/(13))
*
make temp1 end_f2 end_f2
# error4(j)
*
write temp1
*
compute erroresbm2(j,3) = temp1(1,1)
```

```
  *
  write erroresbm2
  *
  *Calculo del error porcentual medio absoluto (MAPE)
  *fuera de la muestra 2004:01-2005:01
  *
  set error(j)        start_f2 end_f2 = abs((iipc_outf - iipc)/iipc)
  set ERR0R2(j)       start_f2 end_f2 = error(j) * 1
  accumalate ERR0R2(j) start_f2 end_f2 error3(j)
  set ERROR4(j)       start_f2 end_f2 = (ERROR3(j)/(13))
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
  *
  compute erroresbm2(j,4) = temp1(1,1)
  *
  write erroresbm2
*
end do j
*
open copy erroresbm2.dat
write(unit=copy) erroresbm2
close copy
*
*halt
*
*******************************************************************************
*Modelo 2: rezagos del IPC(4), BM(24) vs IPC, considerando de 1 a 4 unidades
*escondidas en una superficie escondida fuera de la muestra
*para pronosticar a diciembre de 2005 un paso fuera de muestra
*
seed 10
*
Do j= 1,2
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV1,TRACE,MODE=EPOCH) start 2005:08
  # IIPC{1 TO P} Ibm{1 TO P1}
  # IIPC
  *
  compute array = MV1
  *
  *NNTEST genera la salida (valores ajustados fuera de la muestra)
  *desde el conjunto de*inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST 2005:01 2005:12 array ;*ajuste fuera de la muestra
  # IIPC{1 TO P} Ibm{1 TO P1}
  # iipc_outf

  set actual   2005:01 2005:12 = iipc
  set estimado 2005:01 2005:12 = iipc_outf
  *
  PRINT 2005:01 2005:12 iipc iipc_outf actual estimado
  *
  *
 end do j
*
*halt
*************************************************************
```

```
*
************************************************************
*-------------MODELO ALTERNATIVO 2 (Se incluye M1)---------
************************************************************
*
*MODELACIÓN DE LA RED NEURONAL ARTIFICIAL
*NNLEARN define un nuevo modelo RN y realiza entrenamientos
*adicionales para un modelo existente.Se utiliza semilla.
*
DECLARE RECTANGULAR ERRORESM1(4,4)
DECLARE VECTOR[SERIES] ERROR(4) ERR0R2(4) error3(4) error4(4)
*
COMPUTE P1       = 24
COMPUTE P2       = 9
COMPUTE P3       = 3
seed 10
*
*Modelo 3: rezagos del IPC(4),M1(24) vs IPC, considerando de 1 a 4 unidades (nodos)
*escondidas en una superficie escondida dentro de la muestra
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV1,TRACE,MODE=EPOCH) start end1
  # IIPC{1 TO P} IM1{1 TO p1};* ITC{1 TO p2}
  # IIPC
  *
  compute array = MV1
  *
  *NNTEST genera la salida (valores ajustados) desde el conjunto de
  *inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST start end1 array ;*ajuste dentro de la muestra
  # IIPC{1 TO P} IM1{1 TO p1} ;*ITC{1 TO p2}
  # iipc_out
  *
  set actual   start end   = IIPC
  set estimado start end1 = iipc_out

  *
  PRINT start end1 IIPC iipc_out actual estimado
  *halt
  open plot IPC_M1DM.rgf
  graph(key=below,pattern,HEADER='Figura 6. Pronóstico del IPC con la RNA (dentro de la
muestra), 1990:01-1999:12',$
  subheader ='Modelo alternataivo 2') 2
  # actual    start end1
  # estimado  start end1
  close plot
  *halt
  *
  *Calculo de la raíz del error cuadrático medio (RECM)
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = iipc_out - iipc
  set ERR0R2(j)       start end1 = error(j) * error(j)
  accumalate ERR0R2(j) start end1   error3(j)
  set ERROR4(j)       end1  end1  = (ERROR3(j)/(120-p1))**0.5
  *
  *
  make temp1 end1 end1
```

```
  # error4(j)
  *
  write temp1
  *
  compute erroresM1(j,1) = temp1(1,1)
  *
  write erroresM1
  *
  *Calculo de la raíz del error porcentual cuadrático medio (REPCM)
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = ((iipc_out - iipc)/iipc)
  set ERR0R2(j)       start end1 = error(j) * error(j)
  accumalate ERR0R2(j) start end1 error3(j)
  set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p1))**0.5
  *
  make temp1 end1 end1
  # error4(j)
  *
  write temp1
  *
  compute erroresM1(j,2) = temp1(1,1)
  *
  write erroresM1
  *
  *Calculo del error absoluto medio
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = abs(iipc_out - iipc)
  set ERR0R2(j)       start end1 = error(j) * 1
  accumalate ERR0R2(j) start end1 error3(j)
  set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p1))
  *
  make temp1 end1 end1
  # error4(j)
  *
  write temp1
  *
  compute erroresM1(j,3) = temp1(1,1)
  *
  write erroresM1
  *
  *Calculo del error porcentual absoluto medio
  *dentro de la muestra 1990:01-1999:12
  *
  set error(j)        start end1 = abs((iipc_out - iipc)/iipc)
  set ERR0R2(j)       start end1 = error(j) * 1
  accumalate ERR0R2(j) start end1 error3(j)
  set ERROR4(j)       end1  end1 = (ERROR3(j)/(120-p1))
  *
  make temp1 end1 end1
  # error4(j)
  *
  write temp1
  *
  compute erroresM1(j,4) = temp1(1,1)
  *
  write erroresM1
*
end do j
```

```
*
open copy erroresM1.dat
write(unit=copy) erroresM1
close copy
*
*halt
*
* Estadísticos de bondad de ajuste para el periodo 2000-2002***
*****************************************************************
DECLARE RECTANGULAR ERRORESM11(4,4)
DECLARE VECTOR[SERIES] ERROR(4) ERR0R2(4) error3(4) error4(4)
*
seed 10
*Modelo 3: rezagos del IPC(4)M1(24) vs IPC, considerando de 1 a 4 unidades (nodos)
*escondidas en una superficie escondida dentro de la muestra
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV1,TRACE,MODE=EPOCH) start end1
  # IIPC{1 TO P} IM1{1 TO P1}
  # IIPC
  *
  compute array = MV1
  *
  *NNTEST genera la salida (valores ajustados) desde el conjunto de
  *inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST start_f1 end_f1 array ;*ajuste dentro de la muestra
  # IIPC{1 TO P} IM1{1 TO P1}
  # iipc_out1
  *
  set actual   start_f1 end_f1 = iipc
  set estimado start_f1 end_f1 = iipc_out1
  *
  PRINT start_f1 end_f1 IIPC iipc_out1 actual estimado
  *halt
  open plot IPC_M1DM00-02.rgf
  graph(key=below,pattern,HEADER='Figura 9. Pronóstico del IPC con la RNA (fuera de la
muestra), 2000-2002',$
  subheader ='Modelo alternativo 2') 2
  # actual     start_f1 end_f1
  # estimado    start_f1 end_f1
  close plot
  *halt
  *
  *Calculo de la raíz del error cuadrático medio (RECM)
  *fuera de la muestra 2000-2002
  *
  set error(j)        start_f1 end_f1 = iipc_out1 - iipc
  set ERR0R2(j)       start_f1 end_f1 = error(j) * error(j)
  accumalate ERR0R2(j) start_f1 end_f1   error3(j)
  set ERROR4(j)       end_f1  end_f1  = (ERROR3(j)/(36))**0.5
  *
  *
  make temp1 end_f1 end_f1
  # error4(j)
  *
  write temp1
  *
  compute erroresM11(j,1) = temp1(1,1)
```

```
  *
  write erroresM11
  *
  *Calculo de la raíz del error porcentual cuadrático medio (REPCM)
  *fuera de la muestra 2000-2002
  *
  set error(j)        start_f1 end_f1 = ((iipc_out1 - iipc)/iipc)
  set ERR0R2(j)       start_f1 end_f1 = error(j) * error(j)
  accumalate ERR0R2(j) start_f1 end_f1 error3(j)
  set ERROR4(j)       end_f1  end_f1 = (ERROR3(j)/(36))**0.5
  *
  make temp1 end_f1 end_f1
  # error4(j)
  *
  write temp1
  *
  compute erroresM11(j,2) = temp1(1,1)
  *
  write erroresM11
  *
  *Calculo del error absoluto medio
  *fuera de la muestra 2000-2002
  *
  set error(j)        start_f1 end_f1 = abs(iipc_out1 - iipc)
  set ERR0R2(j)       start_f1 end_f1 = error(j) * 1
  accumalate ERR0R2(j) start_f1 end_f1 error3(j)
  set ERROR4(j)       end_f1  end_f1 = (ERROR3(j)/(36))
  *
  make temp1 end end
  # error4(j)
  *
  write temp1
  *
  compute erroresM11(j,3) = temp1(1,1)
  *
  write erroresM11
  *
  *Calculo del error porcentual absoluto medio
  *fuera de la muestra 2000-2002
  *
  set error(j)        start_f1 end_f1 = abs((iipc_out1 - iipc)/iipc)
  set ERR0R2(j)       start_f1 end_f1 = error(j) * 1
  accumalate ERR0R2(j) start_f1 end_f1 error3(j)
  set ERROR4(j)       end_f1  end_f1 = (ERROR3(j)/(36))
  *
  make temp1 end_f1 end_f1
  # error4(j)
  *
  write temp1
  *
  compute erroresM11(j,4) = temp1(1,1)
  *
  write erroresM11
*
end do j
*
open copy erroresM11.dat
write(unit=copy) erroresM11
close copy
*
```

```
*halt
*
*---------estimación fuera de la muestraIPC_BM 2004:01-2005:01-------
*
DECLARE RECTANGULAR ERRORESM12(4,4)
DECLARE VECTOR[SERIES] ERROR(4) ERR0R2(4) error3(4) error4(4)
*
*Modelo 3: rezagos del IPC(4), BM(24) vs IPC, considerando de 1 a 4 unidades
*escondidas en una superficie escondida fuera de la muestra
*
seed 10
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV1,TRACE,MODE=EPOCH) start 2003:12
  # IIPC{1 TO P} IM1{1 TO P1}
  # IIPC
  *
  compute array = MV1
  *
  *NNTEST genera la salida (valores ajustados fuera de la muestra)
  *desde el conjunto de*inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST start_f2 end_f2 array ;*ajuste fuera de la muestra
  # IIPC{1 TO P} IM1{1 TO P1}
  # iipc_outf

  set actual   start_f2 end_f2 = iipc
  set estimado start_f2 end_f2 = iipc_outf
  *
  *
  PRINT start_f2 end_f2 iipc iipc_outf actual estimado
  open plot IPC_M1FM04-05.rgf
  graph(key=below,pattern,HEADER='Figura 12. Pronóstico del IPC con la RNA (fuera de la
muestra), 2004:01-2005:01',$
  subheader ='Modelo alternativo 2') 2
  # iipc       start_f2 end_f2
  # iipc_outf  start_f2 end_f2
  close plot
  *halt
  *
  *Calculo de la raíz del error cuadrático medio (RECM)
  *fuera de la muestra 2004:01-2005:01
  *
  set error(j)        start_f2 end_f2 = iipc_outf - iipc
  set ERR0R2(j)       start_f2 end_f2 = error(j) * error(j)
  accumalate ERR0R2(j) start_f2 end_f2  error3(j)
  set ERROR4(j)       start_f2 end_f2  = (ERROR3(j)/(13))**0.5
  *
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
  *
  compute erroresM12(j,1) = temp1(1,1)
  *
  write erroresM12
  *
  *Calculo de la raíz del error porcentual cuadrático medio (REPCM)
```

```
  *fuera de la muestra 2004:01-2005:01
  *
  set error(j)        start_f2 end_f2 = ((iipc_outf - iipc)/iipc)
  set ERR0R2(j)       start_f2 end_f2 = error(j) * error(j)
  accumalate ERR0R2(j) start_f2 end_f2 error3(j)
  set ERROR4(j)       start_f2 end_f2 = (ERROR3(j)/(13))**0.5
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
  *
  compute erroresM12(j,2) = temp1(1,1)
  *
  write erroresM12
  *
  *Calculo ded error absoluto medio (MAE)
  *fuera de la muestra 2004:01-2005:01

  *
  set error(j)        start_f2 end_f2 = abs(iipc_outf - iipc)
  set ERR0R2(j)       start_f2 end_f2 = error(j) * 1
  accumalate ERR0R2(j) start_f2 end_f2 error3(j)
  set ERROR4(j)       start_f2 end_f2 = (ERROR3(j)/(13))
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
  *
  compute erroresM12(j,3) = temp1(1,1)
  *
  write erroresM12
  *
  *Calculo del error porcentual medio absoluto (MAPE)
  *fuera de la muestra 2004:01-2005:01
  *
  set error(j)        start_f2 end_f2 = abs((iipc_outf - iipc)/iipc)
  set ERR0R2(j)       start_f2 end_f2 = error(j) * 1
  accumalate ERR0R2(j) start_f2 end_f2 error3(j)
  set ERROR4(j)       start_f2 end_f2 = (ERROR3(j)/(13))
  *
  make temp1 end_f2 end_f2
  # error4(j)
  *
  write temp1
  *
  compute erroresM12(j,4) = temp1(1,1)
  *
  write erroresM12
*
end do j
*
open copy erroresM12.dat
write(unit=copy) erroresM12
close copy
*
*halt
*
*Modelo 3: rezagos del IPC(4), BM(24) vs IPC, considerando de 1 a 4 unidades
```

```
*escondidas en una superficie escondida fuera de la muestra
*para pronosticar hasta diciembre 2005
*
seed 10
*
Do j= 1,4
  *
  NNLEARN(HIDDDEN=j,RSQUAR=0.95,SAVE=MV1,TRACE,MODE=EPOCH) start 2005:08
  # IIPC{1 TO P} IM1{1 TO P1}
  # IIPC
  *
  compute array = MV1
  *
  *NNTEST genera la salida (valores ajustados fuera de la muestra)
  *desde el conjunto de inputs, usando el modelo estimado con NNLEARN
  *
  NNTEST 2005:01 2005:12 array ;*ajuste fuera de la muestra
  # IIPC{1 TO P} IM1{1 TO P1}
  # iipc_outf

  set actual   2005:01 2005:12 = iipc
  set estimado 2005:01 2005:12 = iipc_outf
  *
  *
  PRINT 2005:01 2005:12 iipc iipc_outf actual estimado
  *
end do j
*
halt
```